California Institute of Technology

Analysis and Training — Why Different?

Joseph P. Provenzano
Project Manager, Corps Battle Simulation, JPL
Robert G. Chamberlain
Principal, CBS Modeler, JPL
Timothy Metivier
U.S. Army Combat Developer, NSC

71st Military Operations Research Society Symposium
Marine Corps Base, Quantico, VA
10–12 June 2003

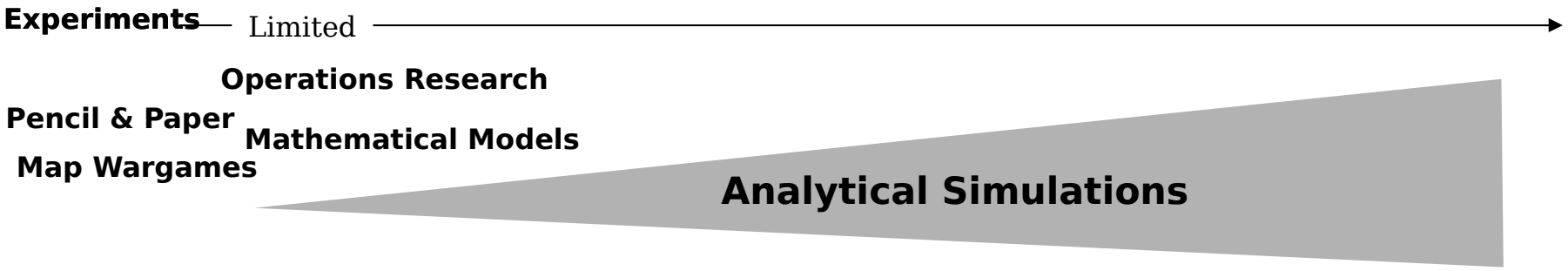
Topics

- Historical Overview
- Credibility: *First* Time and *Every* Time
- Free, Dynamic Play
- COBRA
- Reliability and Recovery (Checkpoint, Modify, Restart)
- Dynamic User Interfaces
- Behavior of Subordinates
- OPFOR
- Gaming the Game
- Magic
- Other Differences
- Conclusions and “Where do we go from here?”

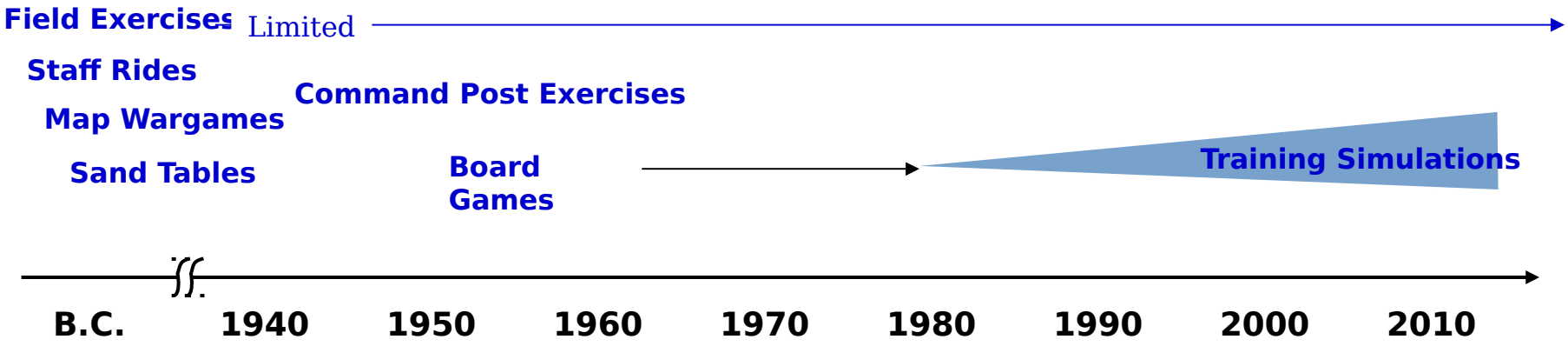
Historical Overview

California Institute of Technology

Analysis

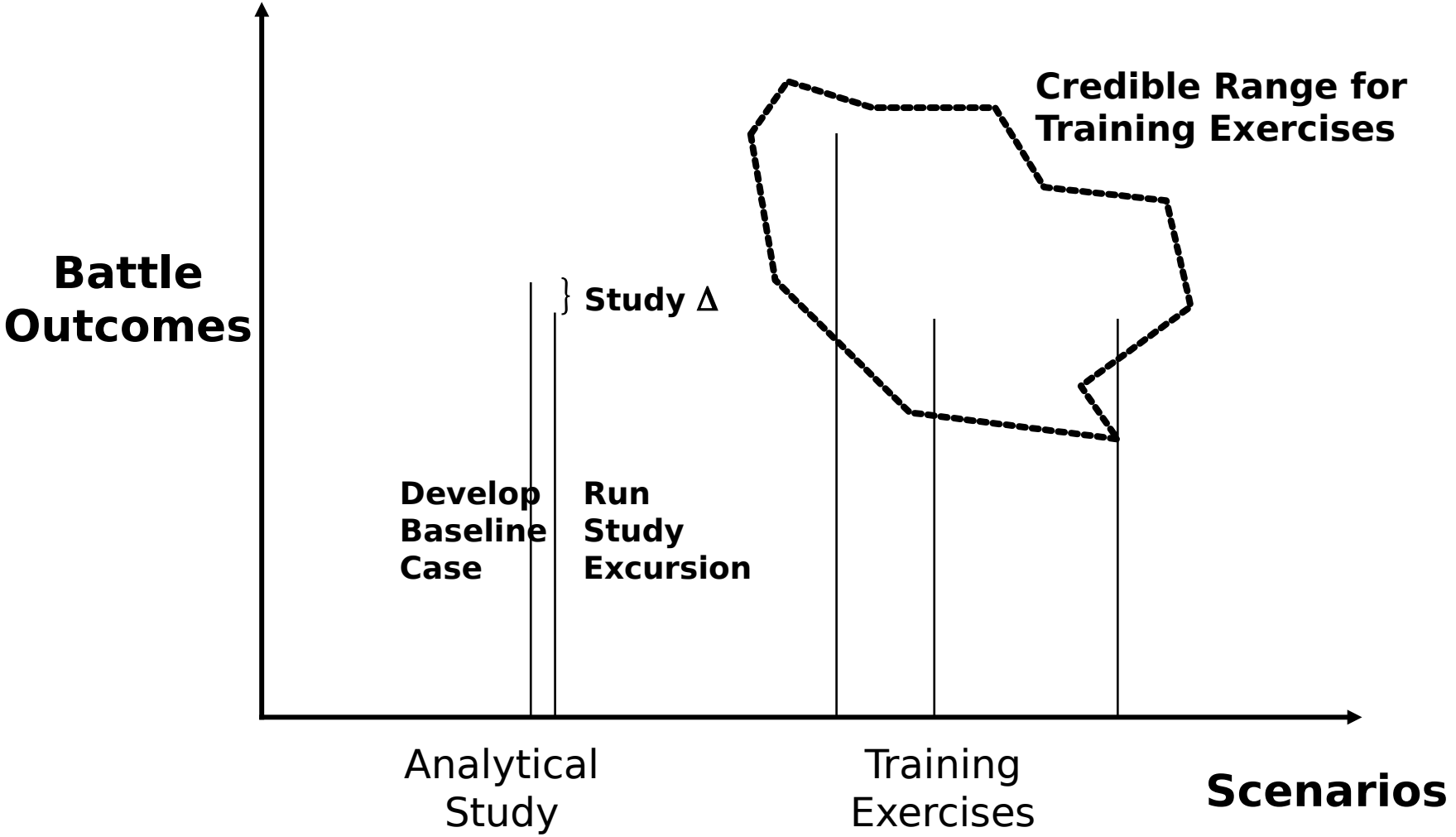


Training



Credibility: *First* Time and *Every* Time

California Institute of Technology



Free, Dynamic Play

Analyses: Controlled quantitative comparisons

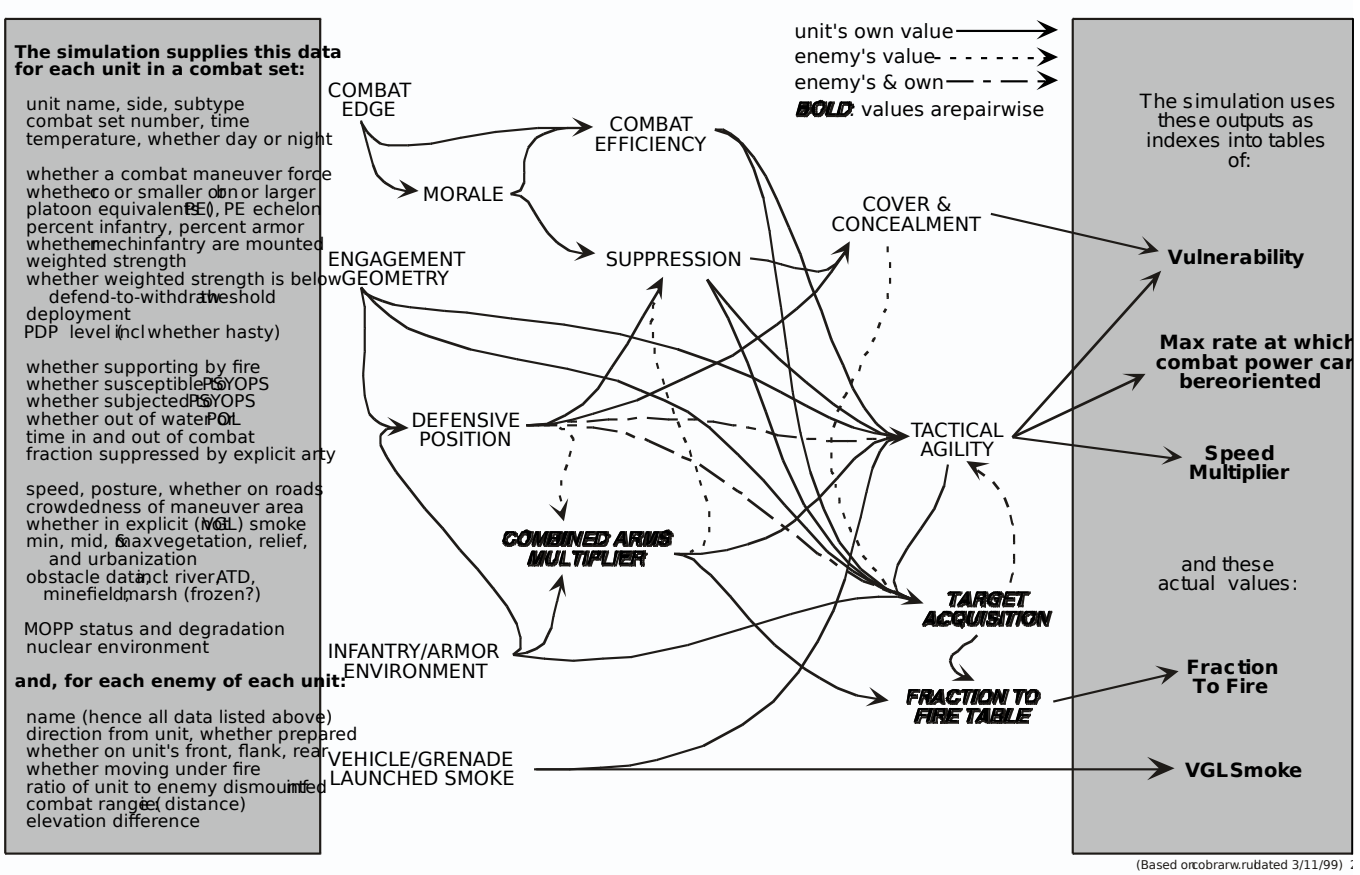
- "Soft" factors undesirable
- Calibrated & tweaked during V&V
- Expect relatively small deltas
- Uncertainties produce statistical comparisons

Training: Chaos

- Action—Reaction—Counteraction
  - » Requires free play with multiple players
- *Qualitative*, not *quantitative*, accuracy is extremely important
  - » Training stimuli must be good enough for decision making
  - » "Soft" factors crucial
  - » "Reward" good decisions with good results
  - » Allow — but "punish" — bad decisions
  - » Unimportant details must be credible

COBRA
Combat Outcome Based on Rules for Attrition

California Institute of Technology



CBS1.8.0 COBRA Rule Set Dependencies

Checkpoint / Modify / Restart
Reliability & Recovery

Recovery from regular checkpoints is essential during training exercises

- Training exercises run at 50+ sites non-stop for days
  - Even if the game stays up, may have “Game Over” too soon
    - » Due to a tactical blunder … or a tactical *coup de grâce!*
  - Even if no code errors, data may contain a “ticking time bomb”
    - » After several days of ops, *any* “computer error” could be a disaster
  - With a half million lines of evolving code, there *will be* errors
    - » Easy recovery is a better goal than impossible error-free code
    - » Also: Memory leaks, performance problems, functional bugs, or … functionality that is unacceptable to users in spite of how *good* it is
  - Either fix the code or disable whole areas of functionality
    - » Full or modular recompilation? How fast *is* that compiler?

Dynamic User Interfaces

Analytical simulations — typically, just the analyst/developer

Training simulations — many different types of users:

- Training audience (with ABCS / C4ISR) and OPFOR (and other sides)
  - » Role players (often know little about computers)
    - ❖ Ground, air, naval, logistics, intelligence, …
  - » Need perceived truth, not ground truth
  - » Free play
- Other software systems
  - » Linked simulations
- Tech control
  - » Take checkpoints, tweak performance parameters, fix problems
- Senior control
  - » Control the exercise, adjudicate disputes, perform “magic”

Behavior of Subordinates

In the real world: subordinates are taught to obey orders *but* to be imaginative in response to unforeseen circumstances

- If the commander is *lucky*, his subordinates will save his … command

In analysis: subordinates should reflect the real world to the extent possible for statistical analysis

In training: subordinates *should not* be imaginative

- *Negative Training would result if subordinate commanders were imaginative and made a bad plan successful — or vice versa**

* Really smart Artificial Intelligence (if it existed) would have to be used very carefully — if at all — in training simulations

OPFOR

OPFOR is a challenge for training simulation developers:

Level playing field is needed but there are many fewer OPFOR controllers controlling a comparable number of units

- Detailed, Symmetrical Modeling would require too many controllers
- Using modeling shortcuts for OPFOR can't preclude training audience options (e.g., resupply convoys must be on the road for targeting)
- Artificial Intelligence support is possible, but can't be perceived as an advantage for OPFOR (e.g., "smart chess player")

Gaming the Game

All players in a training simulation want to *win*

- Incentives are high — sometimes, careers are perceived to be on the line
- To some, an exercise is a *game* whose extended rules include the consequences of violating the nominal rules

In training, there must be no benefit from unrealistic actions
(In the analytical world, analysts don't let weird things happen)

We have seen:

Using a *Withdraw* order to attack—to avoid an automatic threshold

Using differences between maps and the terrain model for ambushes

Filling the battlefield with 2-man OPs because only one enemy could be defeated in a single attrition cycle (thence, CBS has *overrun*)

Magic

Although the *players* must obey the laws of nature, the *exercise director* must be able to "play God" to set the conditions for the commander's training objectives

- Magic resupply / reconstitute
- Magic move
- Magic engineer (mines, PDPs, bridges, on-order obstacles, …)
- Magic NBC (contaminate and decontaminate)
- …
- That is, magically change *any* conditions in the simulation

Magic is not always easy to code—moving a unit in combat could leave fire-support missions with nothing to support, detection processes falsely initialized, and so on…

Bonus: Magic makes testing much easier

Other Differences

Dynamic Task Organization

- Essential capability for training simulations

Scaling

- Multiple echelons are involved in training

Linkage to Other Models

- Real time, stable linkage is needed for training

ABCS / C4ISR Interfaces

- Real time linkage is essential for training

Computer Performance

- Real time performance is required for training

Turnkey Fielding

- Delivered world-wide for training

Summary

When used for analysis, simulations are used like analytical equations

- Analysts control execution and run excursions until the study is complete
- Emphasis is on getting comparative insights based on differences in input parameters

When used for training, simulations are used as if they were a simplified (doctrinal) version of the real world

- Commanders on both sides receive status and intel reports, make free-play decisions, issue orders and respond to real-time outcomes
- Emphases are on reasonable outcomes that reward good and bad command decisions, realistic reporting, real-time execution, and simulation stability

Conclusions

- Using the same simulation for analysis and training is not a realistic goal
- Neither continuing completely separate development nor forcing acceptance of algorithms “thrown over the fence” is the answer
- Need universal recognition that the training community has significantly different requirements and has developed its own legacy of good simulations and useful models

Where do we go from here?

- The M&S community needs a vehicle to begin meaningful dialogue between training and analytical simulation users and developers

- The M&S community needs to strive for a variety of common algorithms for attrition, movement, detection,...
  - “Soft” factors must be controllable
  - “Object-oriented” models?
  - Composable simulations?